On Use of Nav2 Docking

OPEN NAVIGATION

Steve Macenski

Your Friendly Neighborhood Navigator

Owner, Open Navigation

Nav2 Developer, Manager

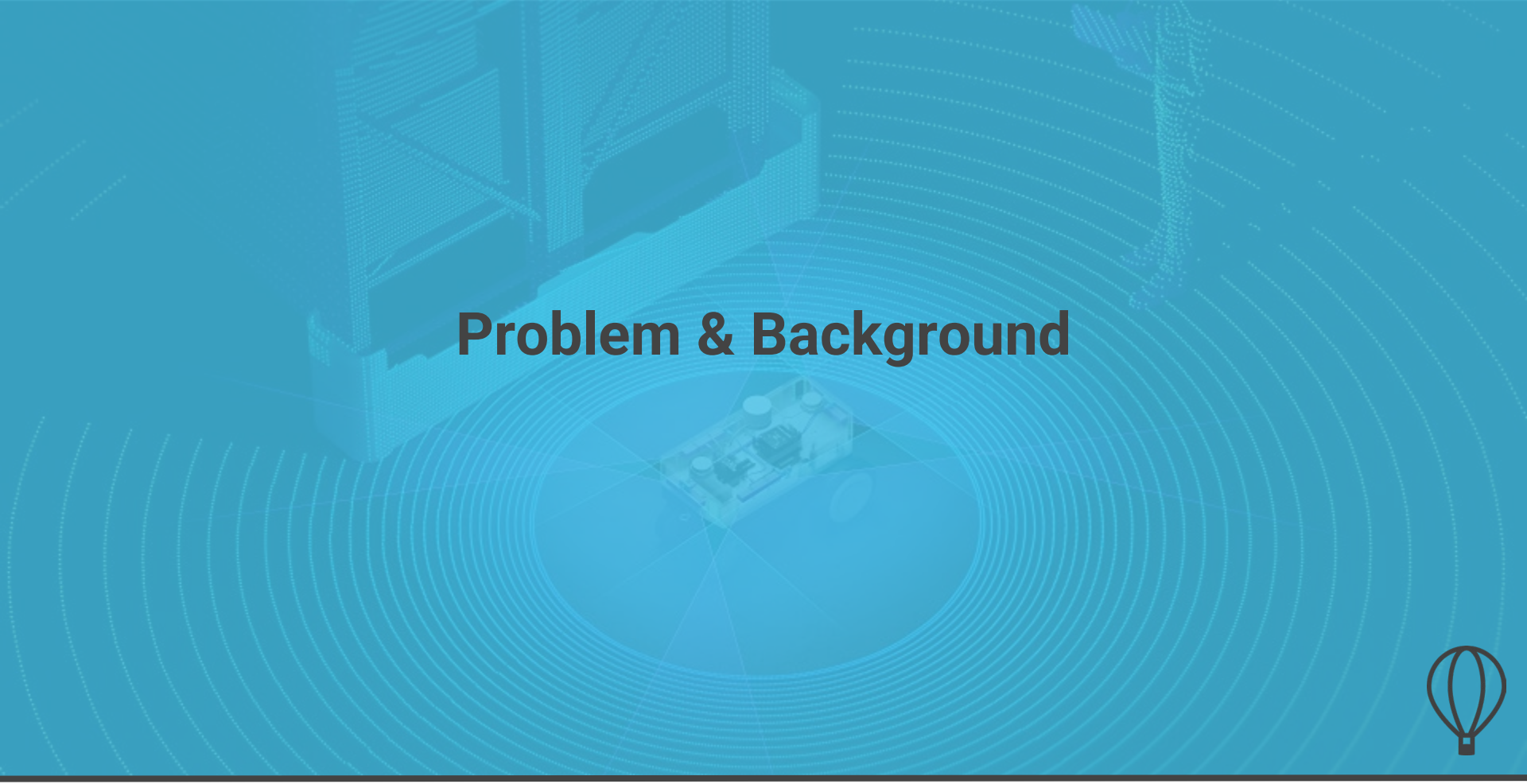
ROS Tech. Gov. Committee

Supports the Robotics Industry via the Democratization of Professional-Quality Robotics Technologies

Offering ROS, Nav2, & Mobility Professional Support Services & Corporate Partnerships

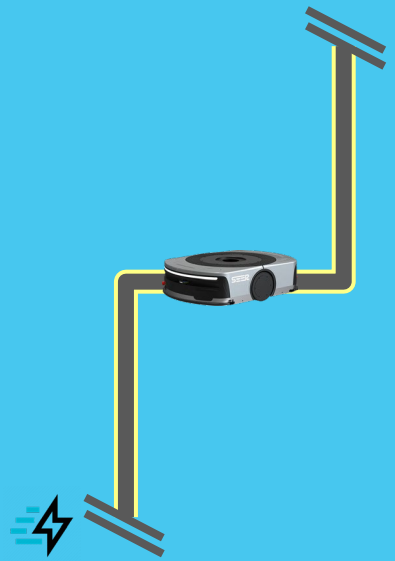
User Supported to Develop & Maintain Key Open-Source Technologies for the Long-Term Future via Sponsorships

Problem & Background



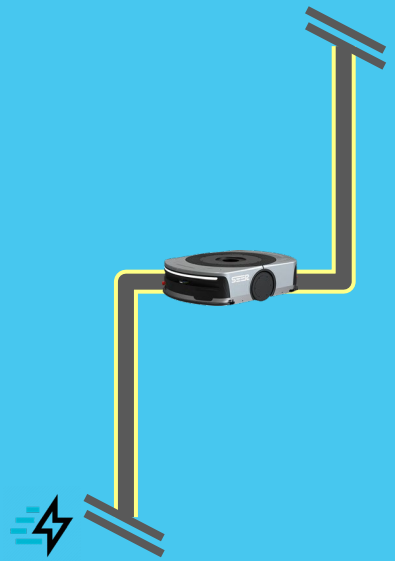
Background - How Do We Keep Our Robots Online?

Background - How Do We Keep Our Robots Online?

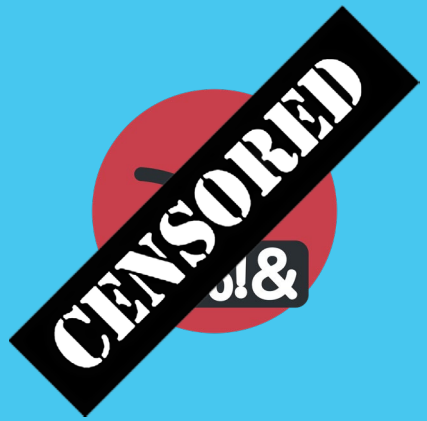


Insitu Charging

Background - How Do We Keep Our Robots Online?

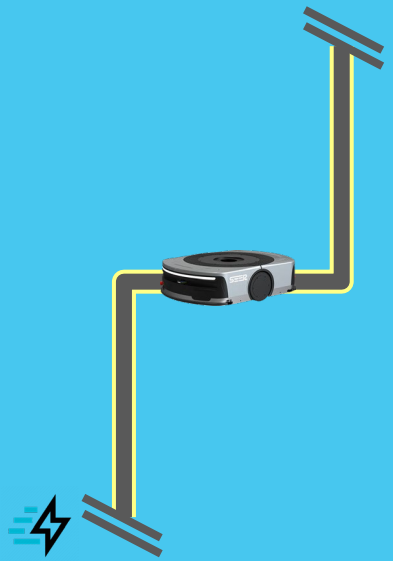


Insitu Charging

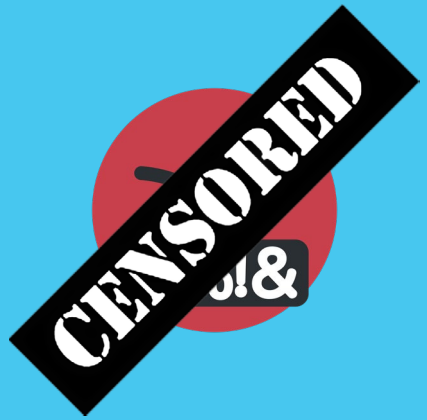


Manual Intervention

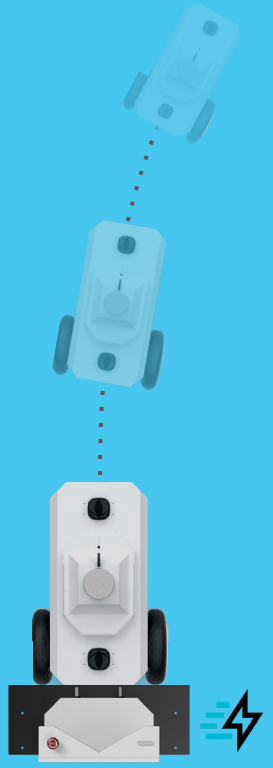
Background - How Do We Keep Our Robots Online?



Insitu Charging



Manual Intervention



Charge-Change Stations

Background - What Options Exist Today?

osrf/autodock

→ Prescribed QR layout, charger interfaces + rudimentary control, but well implemented

ZebraDevs/fetch_open_auto_dock

→ Restrictive licensing + requires Fetch dock, 2D lidar + no feedback, but good control

An array of proof of concept and research projects

All are Unmaintained, ROS 1*, & Support a Single Dock, Sensor, and Layout

Background - What Options Exist Today?

osrf/autodock
→ Presc plemented

ZebraDevs/fet
→ Restrict control

An array of proc

All are Un , Sensor, and Layout

ORDER NOW!!

3 EASY PAYMENTS OF

$39.99

PLUS ONLY $9.99
SHIPPING & HANDLING

AS SEEN ON TV

MasterCard VISA AMERICAN EXPRESS Cards DISCOVER NETWORK

30 DAY MONEY BACK GUARANTEE!

1-800-555-NAV2

NAV2

Our Solution:
Nav2 Docking

A Mature, Fully Generalized, & Supported Docking Solution

→ Independent of Hardware & Detection Method

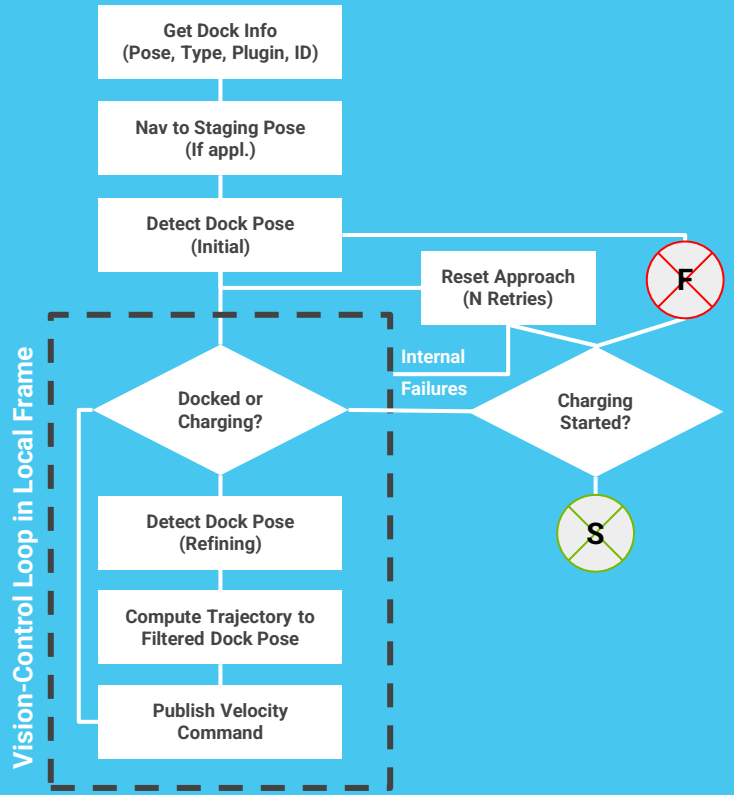
→ Automatic Retry Mechanisms at Each Stage

→ Supports Multiple Dock Locations, Revisions

→ Corrects for Localization Error & Dock Movement

→ Vision-Control Continuously Refines Estimate

→ Plugin Charging & Non-Charging Docks (included)



NAV2

Nav2 Docking - Dock Plugins for Solution Implementation

Plugin API for Application Customization

Two Types: Charging and Non-Charging (for Infrastructure)

Simple {Charging, Non-Charging} Dock Plugins Provided

- → Charging: `BatteryState` charge status or docke
- → Docked: `JointState` torques or proximity
- → Dock Pose: sensor detection or at annotated pose
- → Staging Pose: Offset from dock pose - dist, θ
- → *Sufficient for most users following ROS convention*

Just BYO-Detector! Apriltag, 2D ICP, AI, etc.

```
class ChargingDock
{
  void configure(...)
  void activate()
  void deactivate()
  void cleanup()
  string getName()

  PoseStamped getStagingPose(...)
  bool getRefinedPose(...)

  bool isDocked()
  bool isCharging()
  bool isCharger()

  bool disableCharging()
  bool hasStoppedCharging()
}
```

Contains Database of Dock Instances and Plugins for Large, Heterogeneous Fleets

Nav2 Docking - A Few Auxiliary Details

Detected Features can be Arbitrarily Mis/Aligned with the Dock

Dock Database

→ External yaml, in server config, or in Action request

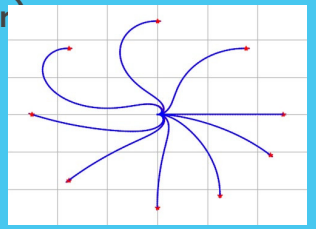
```yaml
# Types of docks
dock_plugins: ['nova_carter_dock']
nova_carter_dock:
  plugin: 'opennav_docking::SimpleChargingDock'
  docking_threshold: 0.05
  staging_x_offset: -0.7
  use_external_detection_pose: true
  use_battery_status: false # true
  use_stall_detection: false

# Dock instances
docks: ['home_dock']
home_dock:
  type: 'nova_carter_dock'
  frame: map
  pose: [0.0, 0.0, 0.0]
  id: 'c67f50cb-e152-4720-85cc-5eb20bd85ce8'
```

Controller

Thanks Alberto Tudela (@ajtudela)!

→ A Smooth Control Law for Graceful Motion (`nav2_graceful_controller`)



The Things You Know and Love About Nav2

→ BT Nodes, Simple Commander API, fully parameterized, tutorials

→ 90% test coverage, control panel, contextual error codes, feedback, ...

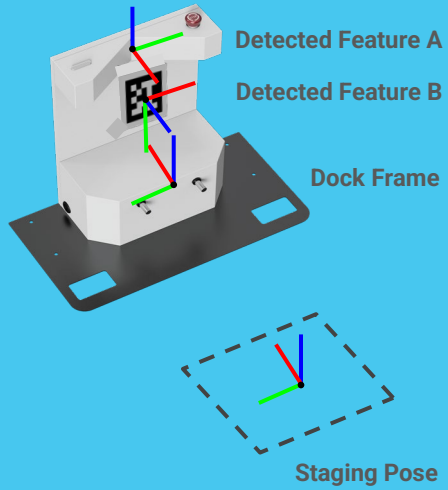
Nav2 Docking - Key Configurations (unexhaustive)

`dock_backwards`
→ Whether docking forward or in reverse

`dock_prestaging_tolerance`
→ Tolerance to staging pose to bypass navigation (if appl.)

`staging_{x,yaw}_offset`
→ Staging pose offset relative to dock location

`external_detection_{translation,rotation}_{x,y,z,r,p,y}`
→ Conversion of feature location into dock location

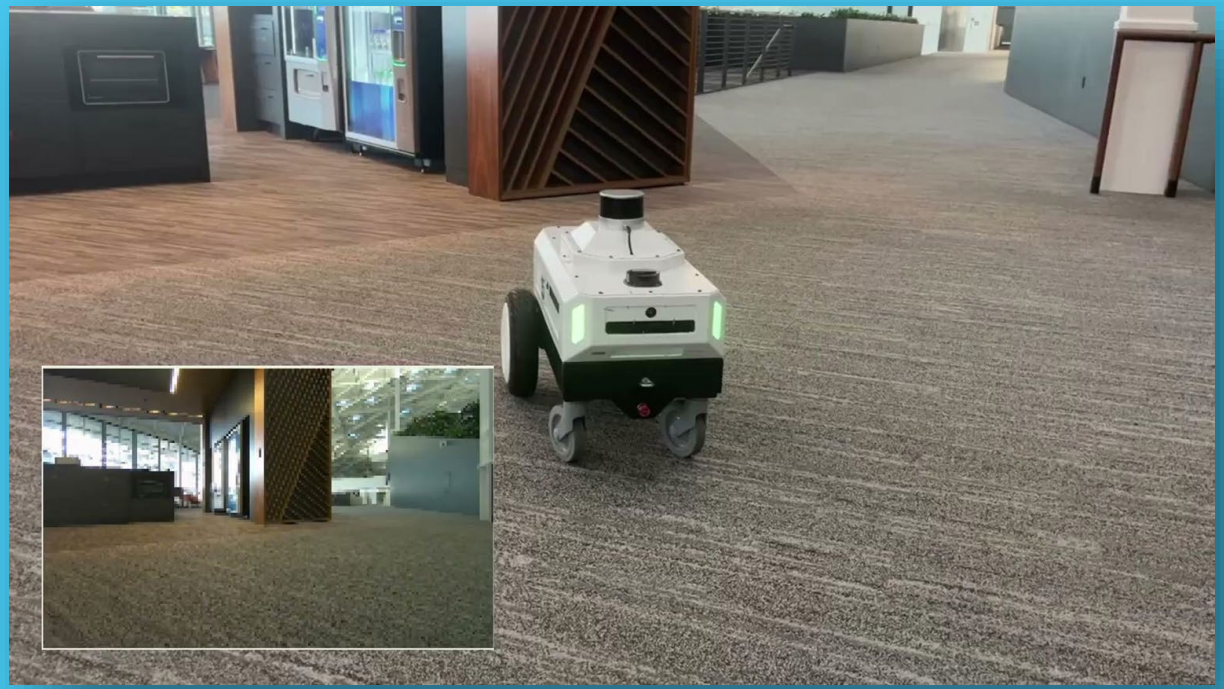


Demonstrations & In the Wild

Demos - Apriltags (Vision) & Duration Testing

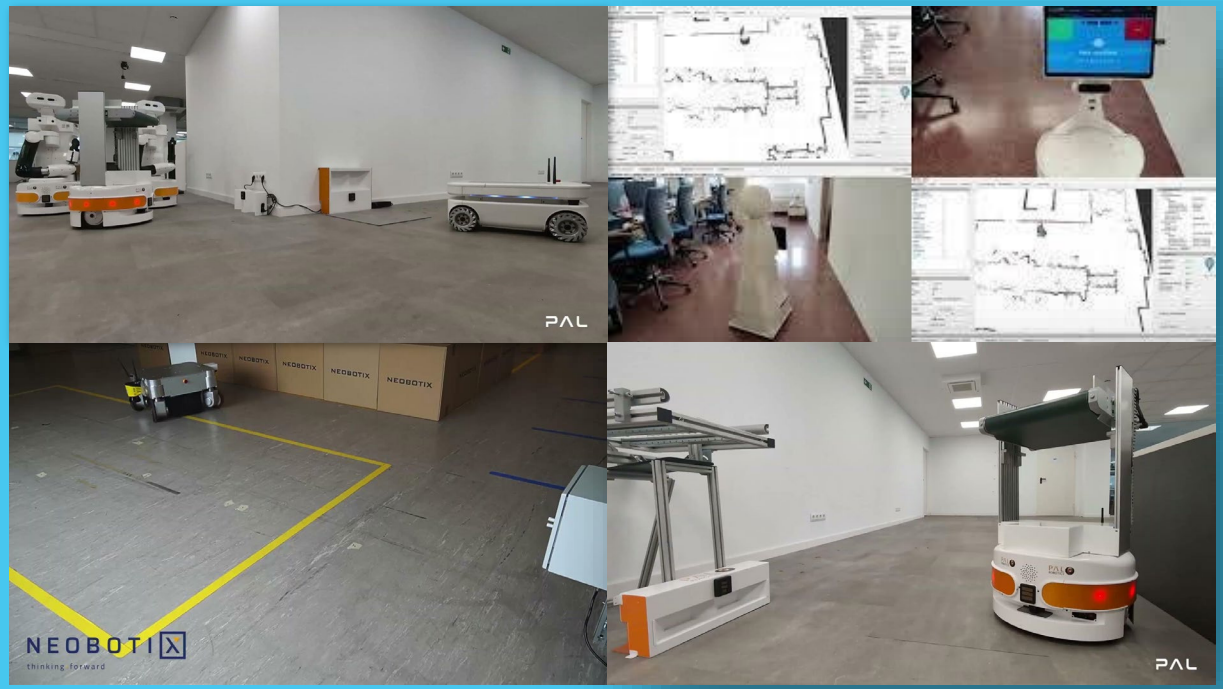
https://github.com/NVIDIA-ISAAC-ROS/nova_carter

Demos - FoundationPose (AI 3D Pose Estimation)



https://github.com/NVlabs/FoundationPose

Demos - ICP Templates (2D Lidar)



Our Sponsors

DEXORY

NVIDIA

POLYMATH ROBOTICS

StereoLabs

CONFIDENTIAL

OPEN NAVIGATION

ROS, Nav2, & Mobility Professional Support Services to Power Your Robotic Solutions

Co-Develop Robust Nav2 Extensions, Specialized Algorithms, Optimizations, AI and Autonomy Systems

Partnerships & Project Sponsorships - addtl resources, publicity, roadmap influence, expert access, & more!

---

nav2.org | opennav.org | github.com/ros-navigation

Steve Macenski

Your Friendly Neighborhood Navigator

Open Navigation

steve@opennav.org

github.com/SteveMacenski